FIATAL MŰSZAKIAK TUDOMÁNYOS ÜLÉSSZAKA

Kolozsvár, 1998. március 20-21.

Védelmi stratégiák a UNIX operációs rendszerben

Somlai Gábor

This paper gives a short overview of the UNIX security components including the critical files managed by the system administrator and the weak points of the user owned ones. It also provides a brief guide of the most common configuring problems and defending strategies. Deals with conventions about SUID programs, trojan horses, including the most frequent tricks carried out by hackers. ETHERNET sniffing, detecting a sniffer program.

A számítógépes hálózatosodás elterjedésével fokozottan előtérbe került a számítógépen tárolt adatok védelme. Az Internetre kapcsolódó számítógépek egyre gyakrabban esnek áldozatul illetéktelen behatolóknak, kiknek célja nem feltétlenül az erőforrás feletti uralom megszerzése (root shell birtoklása, vagy root jogosultságú programok futtatása), hanem sokszor csak egy-egy - amúgy elérhetetlen - állomány letöltése, vagy éppen további támadások indítása más gépek ellen. Ez utóbbi esetben a fő cél a betörő kilétének elrejtése.

A következőkben a biztonsági szempontból kritikus pontokat és az azzal kapcsolatos konvenciókat ismertetem. A lista korántsem lehet teljes, mint ahogy egyetlen védelmi stratégia sem nyújthat 100%-os biztonságot. Azonban a rendszergazdának fel kell vállalni a tökéletességre való törekvést, minthogy a hackerek is újabb és újabb trükköket agyalnak ki.

Kritikus file-ok

Ezen file-ok közös tulajdonsága, hogy kompromittáltságuk esetén a hacker könnyen bejuthat. Ezért az alább említetteken kívül fontos, hogy a tulajdonossági körök és az engedélyek pontosan definiálva legyenek.

- **/etc/passwd**

A legszembetűnőbb, ha ezen file-on történik változás. Triviális, hogy ne legyen új, idegen bejegyzés benne. 0 UID-del rendelkező felhasználók csak a privilegizált felhasználók (root, adm, sys, sync stb.) lehetnek. Egyes rendszerekben az operációs rendszer 0 UID-et rendel a felhasználóhoz akkor is, ha az nem 0, hanem mondjuk 2-nek valamely magasabb hatványa. Ez a getpwent() függvény egy korábbi verziója miatt van. Fontos továbbá, hogy ne legyen két különböző felhasználó ugyanolyan azonosítóval, valamint ne legyen üres a jelszó mező sehol sem.

- **/etc/hosts.equiv**

Meg kell vizsgálni, hogy csak a megbízható host-okat tartalmazza-e (trusted host mechanism), valamint tartalmaz-e "+" jelet.

- **/etc/exports**

Filerendszert csak akkor exportáljunk, ha feltétlen szükséges. Lehetőleg kerüljük az exportálást a világ felé. Ha mégis kell, akkor figyelni kell arra, hogy az exportált jegyzék ne legyen írható.

- **/etc/inetd.conf**

Vizsgáljuk meg, hogy minden szolgáltatáshoz megfelelő program van-e rendelve. Feltűnő, ha a hozzárendelt file egy shell. Meg kell vizsgálni, hogy mindegyik file az eredeti-e, nem trójai faló.

- **.rhosts**

Root esetében a használata kerülendő. Ha már van a root jegyzékében, akkor ne legyen benne "+" jel. Analóg módon a felhasználók .rhosts file-jai se tartalmazzanak "+" jelet. Scriptek rendszeres futtatásával kell ellenőrizni ezen dolgokat.

- **.forward**

Ezen állományokat is rendszeresen kell ellenőrizni, hogy milyen címeket tartalmaznak és tényleg címeket tartalmaznak-e. Ha ebben a file-ban egy pipe van, akkor különösen oda kell figyelni, ugyanis a sendmail a pipe utáni parancsot tekintet nélkül végrehajtja.

- **.*rc file-ok (inicializációs és startup file-ok), .login, .profile, .logout**

Némely program, amely ilyen inicializációs komponenst tartalmaz, rendelkezik olyan képességgel, hogy a felhasználó futás közben kilépjen és shell parancsot hajtson végre. Ezeket az escape szekvenciákat kell kiszűrni a .*rc file-okból. Továbbá olyan fontos környezeti változókat is be lehet itt állítani, mint a PATH és az IFS. Az elérési útvonalnál általános konvenció, hogy a default jegyzék ne legyen a sor elején. Ezzel ugyanis ki lehet küszöbölni, hogy a root a felhasználó által megírt trójai programot hajtson végre pl. ls helyett. A su program használatánál pedig egy óvatos rendszergazdánál elvárható, hogy minden esetben /bin/su -t gépeljen. Egy rosszul beállított IFS pedig azt eredményezheti, hogy a root pl. /bin/ls gépelésénél egy valahol elhelyezett trójai bin programot indít el ls argumentummal (IFS="/" esetén). Az umask shell parancs is ilyen file-okban szokott végrehajtódni, ezért ezek beállítását is ellenőrizni kell.

- **Az utmp és wtmp naplózási file-ok**

Az utmp a bejelentkezettek listáját tartalmazza és egyszerűen megnézhető a w paranccsal. Így ez, valamint az lsof (list open files) használatával pontosan megállapítható a felhasználó jelenlegi tevékenysége. Ez utóbbi file nem része a standard UNIX-nak, de sok helyről ingyen letölthető. A wtmp az előző bejelentkezéseket tartalmazza és a last paranccsal listázható, így ki lehet szűrni ha egy "felhasználó" kívülről lép be.

Jogosultságok

- **SUID bit**

A rendszergazda rutinfeladatai közé tartozik a setuid file-ok vizsgálata. Akár a cron-ba beillesztve, de rendszeresen meg kell vizsgálni ezeket a programokat. Ugyanis a már bejutott hacker következő lépése az, hogy egy rejtekajtót telepítsen a rendszeren a későbbi visszatéréshez. Ennek során gyakran új SUID file-ok jelennek meg a rendszeren. Kézenfekvő dolog egy SUID shell létrehozása, azonban egyes rendszereken ez ellen az operációs rendszernek beépített védelme van. Csakúgy, mint a SUID scriptek esetében. Ugyanis egy könnyen átlátható struktúrájú scriptnél - ellentétben a bináris programokkal - könnyen lehet úgy módosítani a futási környezetet, hogy ugyanaz a program teljesen másként működjön.

Ha mégis vannak SUID scriptek, azok a

/etc	/bin	/sbin	/usr/sbin
/usr/etc	/usr/adm	/usr/lib	/usr/X11R*
/usr/local/etc	/usr/local/bin		

jegyzékekben találhatók. Ha ezeken kívül találkozunk SUID programokkal, akkor joggal feltételezhetjük, hogy kompromittálták a gépet. Ezért érthető, hogy egy tapasztaltabb hacker jól elrejtje őket. Legtöbbször rejtetté teszi, vagy akár egy egész rejtett jegyzékbe teszi azt. Ilyen módon tehát a felhasználónál lévő .xx vagy .mail jegyzék erősen gyanús lehet. Másik stratégia, hogy a file neve ne legyen szem előtt: pl. ".. " (pont, pont, szóköz) vagy "..." . Gyakran szoktak vezérlő karaktereket a nevekben ("..^G"). Azonban ezeket egy tapasztalt rendszergazda könnyedén hatástalanítja.

Trójai faló

A trójai falóvak olyan programok, amelyek már meglévő - és gyakran a UNIX részét képező - programok mintájára készültek és csak látszólag működnek úgy, mint az eredeti. Így tehát például egy trójai login elsődleges célja nem a felhasználó beléptetése, hanem a felhasználó-jelszó páros megszerzése, miközben természetesen végre is hajtja eredeti feladatát, teljesen szokványos módon. De akkor honnan lehet felismerni őket? Először is a hossza gyakran nem egyezik az eredeti file-éval. Ha tudjuk az eredeti hosszt,

akkor könnyen felismerhetjük a hamist. Ehhez nyújtanak segítséget a különböző file-integritásellenőrző segédeszközök (pl.:TRIPWIRE). Ezek különböző kódoló algoritmusok segítségével mintegy ujjlenyomatot készítenek a filerendszer egészéről és időközönként ellenőrzik azt.

Snifferek

A hálózaton szaglászó programokat szokták így hívni. A hálózati interfész promiszkuis módra állításával ugyanis elérhető, hogy a hálózaton keringő összes csomagot érzékelje. Ez a rendszergazdának hasznos eszköze lehet a hálózati adatforgalom elemzésében. Azonban mivel a legtöbb ETHERNET hálózaton a bejelentkezés során a felhasználói nevek és a jelszavak is kódolatlanul továbbítódnak, könnyen lehet az ilyen programokat ezek megszerzésére használni. A hálózati interfészt csak root állíthatja át, tehát ha ilyet futtat valaki, akkor biztosak lehetünk benne, hogy az adott gépen már rendszergazdai jogkörrel bír. Hogyan lehet egyáltalán észrevenni? Egyik jele a sniffer háttérben való futásának a lassú számítógép és az ezáltal valamelyest lelassuló hálózati adatforgalom. Másik jele lehet maga a futó program. Bár ha valaki ravasz akar lenni, akkor olyan néven fogja futtatni, ami nem gyanús. Azonban ezt is ki lehet küszöbölni a már említett lsof programmal. Egyébként a promiszkuis módon működő hálózati kártya csak úgy ismerhető fel, ha egyenként ellenőrizzük mindet (netstat -r).

Somlai Gábor (hallgató)

Miskolci Egyetem, Informatikai Intézet, 3515 Miskolc-Egyetemváros

Tel.: (0036) 46 365111